Learning and Adaptation for Intelligent Robot

Toshio Fukuda*   Yasuhisa Hasegawa**
* Center for Cooperative Research in Advanced, Science and Technology, Nagoya University
**Dept. of Micro System Engineering, Nagoya University
Furo-cho, Chikusa-ku, Nagoya 464-8603, Japan
Keywords: Learning, Adaptation, Intelligence, Neural Network, Fuzzy Inference, Evolutionary Computation

Abstract - Intelligent systems are required in knowledge engineering, computer science, mechatronics and robotics. This paper discusses the machine(system) intelligence from the viewpoints of learning and adaptation of creatures. Next, this paper introduces computational intelligence including neural network, fuzzy system, and genetic algorithm. Finally, this paper shows some examples of intelligent robotic system: brachiation robot and four-fingered robot hand.

1 Introduction

Intelligence for robot to grow and evolve can be observed both through growth in computational power, and through the accumulation of knowledge of how to sense, decide and act in a complex and dynamically changing world. There are four elements of intelligence: sensory processing, world modeling, behavior generation and value judgement. Input to, and output from, intelligent system are via sensors and actuators. Recently, intelligent systems have been discussed in knowledge engineering, computer science, mechatronics and robotics. Various methodologies about intelligence have been successfully developed.

Artificial intelligence (AI) builds an intelligent agent, which perceives its environment by sensors, makes a decision and takes an action [1]. McCulloch and Pitts suggested that suitably defined networks could learn [2], and furthermore, Newell and Simon developed general problem solver [1]. Afterward, knowledge-based system including expert systems has been developed [1]. In addition, language processing, reasoning, planning, and others have been discussed in AI so far [1]. Human language enables the symbolic processing of information, and is translate into numerical information according to an objective, that is, word is classified into a certain attribute out of much information. In this way, the symbolic information processing have resulted in success in AI. Further, The recent research fields concerning intelligence, include brain science, soft computing, artificial life and computational intelligence [1-6].

Computational intelligence from the viewpoints of biology, evolution and self-organization tries to construct intelligence by internal description, while classical AI tries to construct intelligence by external (explicit) description. Therefore, information and knowledge of a system in computational intelligence should be learned or acquired by itself.

Robot is required to have intelligence and autonomous ability when it works far from an operator with large time delay, or when it works in a world containing ambiguous information. The robot collects or receives necessary information concerning its external environment, and takes actions to the environment. Both of them are often designed by human operators, but ideally, the robot should automatically perform the given task without human assistance. Computational intelligence methods including neural network (NN), fuzzy logic (FL) and evolutionary computation (EC), reinforcement learning, expert system and others, have been applied to realize intelligence on the robotic systems [2-8]. In addition, behavior-based AI has been discussed as learning methods dependent on environmental information [1,9]. The behavior-based AI stresses the importance of the interaction between robot and environment, while classical AI is based on the representation and manipulation of explicit knowledge. Recently, behavior analysis and training as methodology for behavior engineering and model-based learning, have been proposed [9]. In this paper, we introduce a basic technique to build an intelligent system. After that, we introduce adaptation algorithm for brachiation robot and evolutionary computation for a four-fingered robot hand.

2 Intelligent System

In future, a robot will work out of a factory, in which an environment was simplified so that a robot could recognize it. The robot is required to have intelligence and autonomous capability when it works far from an operator with large time delay such as tele- operation, when sensing informations are contained ambiguous information. Key technologies for system intelligence and autonomous are knowledge representation, recognition, inference, search, planning, learning, prediction and so on [1].

The system intelligence emerges from the synthesis of various intelligent capabilities of the systems. Consequently, the whole intelligence of a system depends on the structure for processing information on hardware and software, and this means that the structure determines the potentiality of intelligence [10]. Therefore, we should consider a whole structure of intelligence for information process flow over the hardware and software.

3 Computational Intelligence

3.1 Neuro-Computing and Fuzzy Computing

Artificial neural network and fuzzy logic inference are based on the mechanism and information process of human brain. The human brain processes information super-quickly. McCulloch and Pitts proposed that a suitably defined network could learn in 1943 [2]. After that, the rediscovery of back-propagation algorithm by Rumelhart, popularized artificial NN [2]. The artificial NN simulating the biological brain can be trained to recognize patterns and to identify incomplete patterns. The basic attributes of NN are the architecture and the functional properties; neurodynamics. The neurodynamics plays the role of non-linear mapping from input to output. NN is composed of many interconnected neurons with input, output, synaptic strength and activation.

The learning algorithms for adjusting weights of synaptic strength are classified into two types: supervised learning with target responses and unsupervised learning without explicit target responses. In general, a multi-layer NN is trained by a back propagation algorithm based on the error function between the output response and the target response. However, the back propagation algorithm, which is known as a gradient method, often misleads to local minimum. In addition, the learning capability of the NN depends on the structure of the NN and initial weights of the synaptic strength. Therefore, the optimization of the structure and the synaptic strength is very important for obtaining the desired target response. The other artificial NNs are Hopfield network, Boltzmann Machine, Adaptive Resonance Theory and Self-Organizing Map [2]. The Hopfield network is regarded as an autoassociative fully connected network which has symmetrically weighted links [2]. The Boltzmann machine is based on the simulated annealing according to Metropolis dynamics [2]. The adaptive resonance theory model, which was developed by Grossberg and Carpenter, is composed of input/comparison layer and output/recognition layer [7,11]. The self-organizing map, which was proposed by Kohonen, is a clustering algorithm creating a map of relationships among input and output patterns [7,12].

While NN simulates physiological features of human brain, fuzzy logic inference simulates psychological features of human brain. Fuzzy logic provides us the linguistic representation such as 'slow' and 'fast' from numerical value. Fuzzy logic [4,5,7] expresses a degree of truth, which is represented as a grade of a membership function. It is a powerful tool for non-statistic and ill-defined structure. Fuzzy inference system is based on the concept of fuzzy set theory, fuzzy if-then rule, and fuzzy inference. The fuzzy inference derives conclusions from a set of fuzzy if-then rules. Fuzzy inference system implements mapping from its input space to output space by some fuzzy if-then rules. The widely used fuzzy inference systems are Mamdani fuzzy models and Takagi-Sugeno fuzzy models, which are used as a fuzzy controller. The feature of the fuzzy controller is the locality of control and the interpolation among local control laws. In the fuzzy controller, the state space of the system is divided into some regions as membership functions which are antecedent part, and the output (consequence) for the system control is designed as singletons, linear functions or membership functions. Next, the fuzzy rules are interpolated as a global controller. From the viewpoint of calculation in the inference, inference types are classified into min-max-gravity method, product-sum-gravity method, functional fuzzy inference method and simplified fuzzy inference method. In order to tune fuzzy rule, delta rule has been often applied to the functional fuzzy inference method and to the simplified fuzzy inference method like fuzzy-neural networks.

3.2 Evolutionary Computing

Evolutionary computation (EC) is a field of simulating evolution on a computer [7]. From the historical point of view, the evolutionary optimization methods can be divided into three main categories, genetic algorithm (GA), evolutionary programming (EP) and evolution strategy (ES)

[2,3,7,8]. These methods are fundamentally iterative generation and alternation processes operating on a set of candidate solutions, which is called a population. All the population evolves toward better candidate solutions by selection operation and genetic operators such as crossover and mutation. The selection operation picks up better solutions for the next generation, which limits the search space spanned by the candidate solutions. The crossover and mutation generate new candidates. EC methods can be divided into several categories from various points of view. This paper divides EC methods into genetic algorithm (GA) and evolutionary algorithm (EA) from the representation level .

GAs use simple symbolic operations from the viewpoint of genotype, and GAs are often applied to combinatorial optimization problems such as knapsack problems, traveling salesman problems and scheduling problems [7,8]. It is experimentally known that the GAs can obtain near or approximately optimal solutions with less computational cost. Other GAs are genetic programming and classifier system. The genetic programming, which was proposed by Koza [8] can deal with the tree structure and have been applied for generating computer programs. The classifier system, which is known as a GA-based machine learning method, can learn syntactically simple string rules to guide its performance in an arbitrary environment.

On the other hand, EAs use numerical operations from the viewpoint of phenotype, but EAs also use symbolic operation such as mutation and crossover. EAs including EP and ES, have been often applied for solving numerical optimization problems such as function optimization problems, weight optimization of NN [7]. The important feature of EAs is the self-adaptation, especially self-adaptive mutation is very useful operation. The search range can be adjustable according to its performance [7]. In the EAs, tournament selection and deterministic selection are often applied as the selection scheme.

In addition, the ECs provide the evolution mechanism for population dynamics, robot society and A-life [6]. From the viewpoint of simulated evolution, GAs can maintain the genetic diversity in a population to adapt to dynamic environment. Therefore, ECs are often called adaptive systems. However, the ECs eliminate worse individuals from the population only according to the evaluation from the current environment. As a result, it is difficult that the population adapts to a big change of the environment. Therefore the ECs often require methods to maintain genetic diversity in a population for the dynamically changing environment.

3.3 Synthesized Approach

To realize higher intelligent system, a synthesized algorithm of various techniques is required. Figure 1 shows the synthesis of NN, FL and EC. Each technique plays the peculiar role for intelligent function. There are not complete techniques for realizing all features of intelligence. Therefore, we should integrate and combine some techniques to compensate the disadvantages of each technique. The main characteristics of NN are to classify or recognize patterns, and to adapt itself to dynamic environments by learning, but the mapping structure of NN is a black box and incomprehensible. On the other hand, FL has been applied for representing human linguistic rules and classifying numerical information into symbolic class. It also has reasonable structure for inference, which is composed of if-then rules like human knowledge. However FL does not fundamentally have the learning mechanism. Fuzzy-neural networks have developed for overcoming their disadvantages [5]. In general, the neural network part is used for its learning, while the fuzzy logic part is used for representing knowledge. Learning capability is fundamentally performed as necessary change such as incremental learning, back propagation method and delta rule based on error functions. EC can also tune NN and FI. However, evolution can be defined as resultant or accidental change, not necessary change, since the EC can not predict and estimate the effect of the change. To summarize, an intelligent system can quickly adapt to dynamic environment by NN and FI with the back propagation method or delta rule, and furthermore, the structure of intelligent system can globally evolve by EC according to the objective problems. The capabilities concerning learning adaptation and evolution can construct more intelligent system. Intelligence arises from the information processing on the linkage of perception, decision making and action.

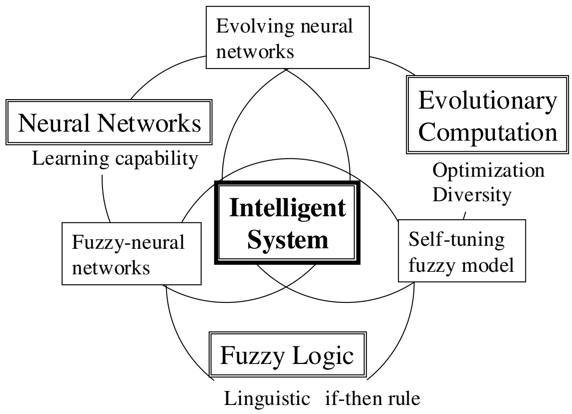

Fig. 1 Synthesis of NN, FI and EC

4 Intelligent robotic system

4.1 Brachiation robot

The brachiation robot is a mobile robot, which dynamically moves from branch to branch like a gibbon, namely long-armed ape, swinging its body like a pendulum[13][14](Fig.2). A lot of research about a brachiation type locomotion robot had been carried out. Saito et al[15] proposed the heuristic learning method for generating feasible trajectory for two-link brachiation robot. Fukuda et al[16] propose the self-scaling reinforcement learning algorithm to generate feasible trajectory with robust property against some disturbances. The reinforcement learning method builds a fuzzy logic controller with four inputs and one output. In these studies, the controller is acquired in a try-and-error learning process and a dynamics model of the two-link brachiation robot is not used for controller design. On the other hand, Nakanishi et al,[17] took another approach, using target dynamics, for controlling an underactuated systems. The two-link brachiation robot is an underactuated system with two degrees of freedom and one actuator. As a two-dimensional extended model, seven-link brachiation robot is studied by Hasegawa et al[18]. The seven-link brachiation robot has redundancy to locomote so that it is able to take a dexterous motion like a real ape in plane, however a dynamical locomotion robot with multi-degree of freedoms is difficult to be controlled. A hierarchical behavior architecture is adopted to design the controller with multi-input and multi-output efficiently. The behavior controllers and their coordinators in the hierarchical structure are generated using reinforcement learning method. The concept of the hierarchical behavior controller is based on the behavior-based control, which has an advantage of designing the controller for a higher-level behavior of the complex system from simpler behaviors in reasonable process.

We developed 13-link brachiation robot shown in fig. 3, that has almost same dimensions and weight as a real long-armed ape. The hierarchical behavior controller shown in fig. 4 generates dynamical motion controlling 14 actuators. Hasegawa et al[19] proposed an adaptation algorithm for brachiation behavior in order to locomote on different branch intervals successfully and achieve continuous locomotion(Fig.5). This adaptation algorithm adjusts four coefficients from behavior coordinator “locomotion” to four behavior controllers using Newton Raphson method when the branch intervals are extended from 90cm to 101cm. In order to achieve the continuous locomotion, the end posture of first swing should be useful for the second swing. We therefore applied the adaptation algorithm to tune the secondary swing motion controller with two parameters.

4.2 Regrasping motion of four-fingered robot hand

Multi-fingered robot hand has an advantage to change the object posture during grasping as well as to grasp various shapes of objects. However, planing of this regrasping motion is hard task, because of a lot of parameters to be determined: grasping points, regrasping points, the object posture at regrasping moment, grasping force and grasping finger to be used.

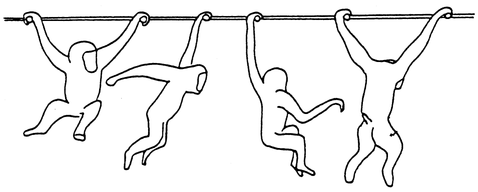

Fig. 2 Brachiation motion of a long-armed ape

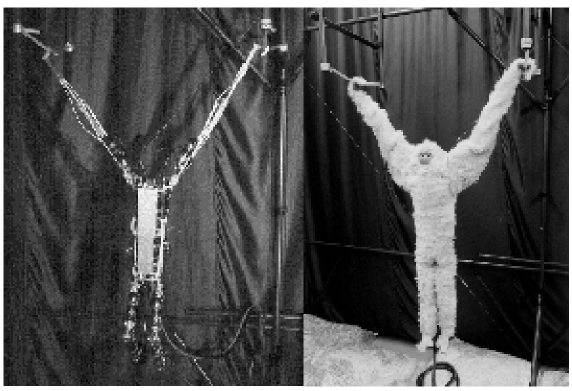

Fig. 3 13-link brachiation robot

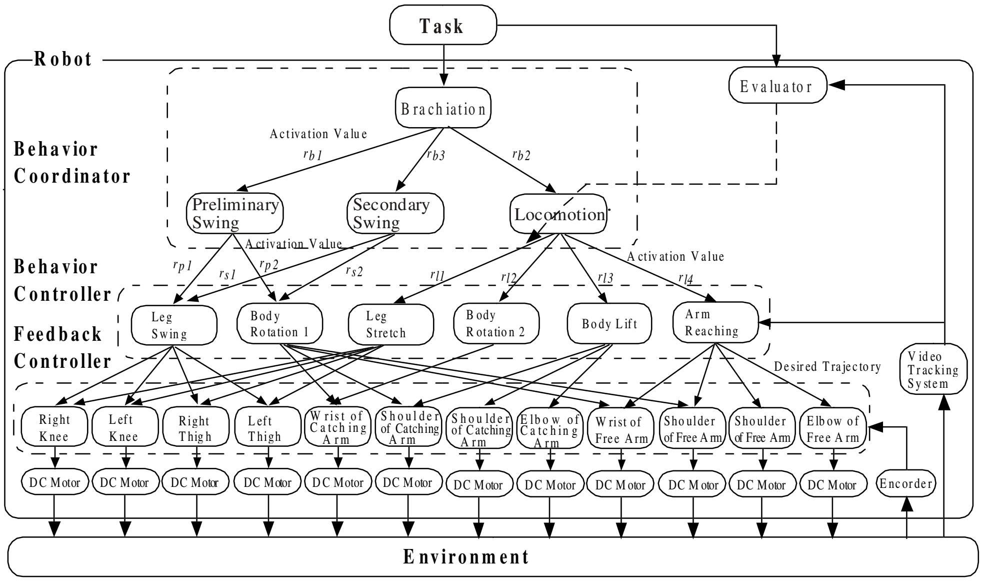

Fig. 4 Behavior-based controller for brachiation robot

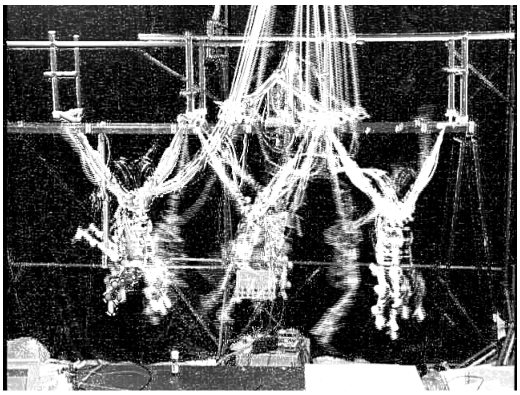
Fig. 5 Continuous locomotion

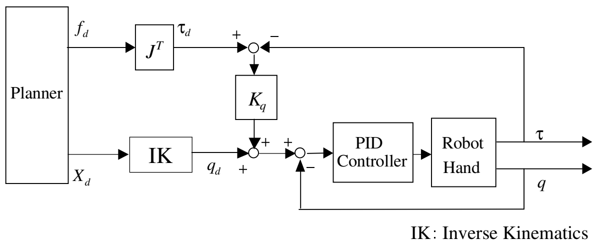

Fig. 6 Block Diagram

We proposed the algorithm to generate a regrasping motion using evolutionary programming(EP) [20]. EP is more effective to find a numerical solution like a grasping point than genetic algorithm. What we have to determine is the initial posture, the final posture of the grasping object and regrasping times. Evolutionary computation requires much iteration until it finds the solution, therefore, the regrasping motion is generated in numerical simulation. The obtained regrasping strategy is applied to the real robot system. Figure 6 shows the block diagram, in which planner means the desired grasping forces and the desired grasping points. The regrasping motion is shown in fig. 7.

5 Summary

This paper presented recent research fields concerning computational intelligence. The computational intelligence is including neural network, fuzzy logic and evolutionary computation. The synthesis of neural network, fuzzy logic and evolutionary computation is important for advanced information processing and structure optimization. Furthermore, this paper showed the brachiation robot as an example of adaptation algorithm for hierarchical behavior-based control architecture, and the four-fingered robot hand as an application example of evolutionary computation. Intelligence of both robots is limited into motion generation capability. They do not deal with perception and of decision making. As future work, we should study about intelligent perception and decision algorithm from ambiguous information and stored acquired knowledge.

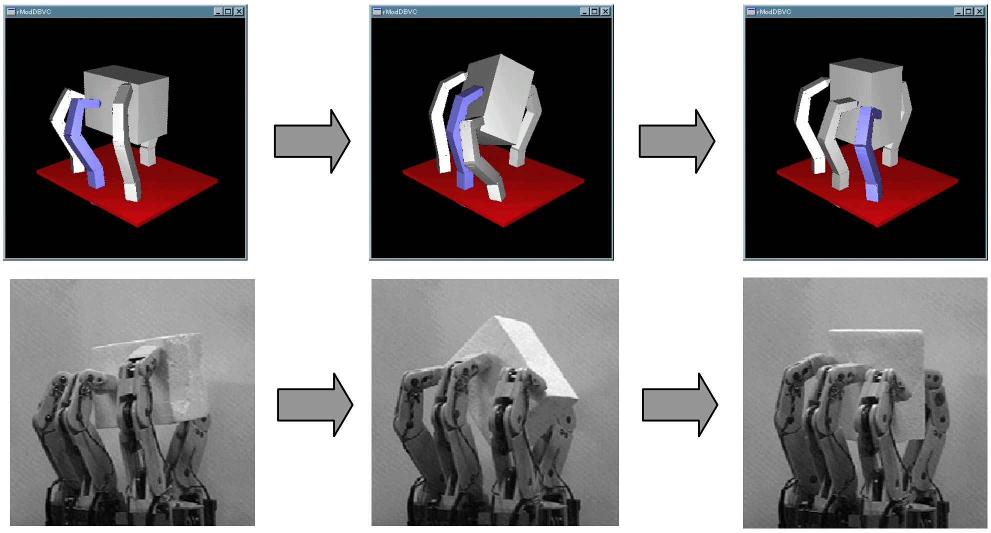

Fig. 7 Regrasping motions (Numerical simulation results and experimental results)

References

[1]S.J.Russell, P.Norvig, Artificial Intelligence, Prentice-Hall, Inc. (1995).

[2]J.M.Zurada, R.J.Marks II, C.J.Robinson, Computational Intelligence - Imitating Life, IEEE Press (1994).

[3] M.Palaniswami, Y.Attikiouzel, R.J.Marks II, D.Fogel, T.Fukuda, Computational Intelligence - A Dynamic System Perspective, IEEE Press (1995).

[4]L.A.Zadeh, Fuzzy Sets, Information and Control, Vol.8, 338-353 (1965).

[5]J.-S.R.Jang, C.-T.Sun, E.Mizutani, Neuro-Fuzzy and Soft Computing, New Jersey: Prentice-Hall, Inc. (1997).

[6]C.G.Langton, Artificial Life -An Overview, The MIT Press (1995).

[7]D.B.Fogel, Evolutionary Computation, IEEE Press (1995).

[8]J.Koza, Genetic Programming II, The MIT Press (1994).

[9]R.Brooks, A Robust Layered Control System for a Mobile Robot, IEEE Journal of Robotics and Automation, RA-2-1, 14-23 (1986).

[10]T.Kohonen, Self-Organization and Associative Memory, Springer-Verlag (1984).

[11]T.Fukuda, N.Kubota, Computational Intelligence in Robotics and Mechatronics, Proc. of The 23rd Int. Conf. on Industrial Electronics, Control, and Instrumentation, 1517-1528 (1997).

[12]T.Fukuda, N.Kubota, Intelligent Robotic System -Adaptation, Learning and Evolution, Proc. of The Third Int Symposium on Artificial Life and Robotics 40-45 (1998).

[13]T. Fukuda, H. Hosokai and Y. Kondo, Brachiation Type of Mobile Robot, Proc. IEEE Int. Conf. Advanced Robotics, pp. 915-920,(1991).

[14]Spong, M. W., Swing Up Control of the Acrobot, Proc. 1994 IEEE Int. Conf. on Robotics and Automation, pp. 2356-2361, (1994)

[15]F. Saito, T. Fukuda, F. Arai, Swing and Locomotion Control for a Two-Link Brachiation Robot, IEEE Control Systems, Vol. 14, No. 1, pp. 5-12, (1994).

[16]T. Fukuda Y. Hasegawa K. Shimojima and F. Saito, Self Scaling Reinforcement Learning for Fuzzy Logic Controller, IEEE Int. Conf. on Evolutionary Computation, pp. 247-252, Japan, (1996)

[17]J. Nakanishi, T. Fukuda and D.E. Koditschek, Experimental implementation of a Target dynamics controller on a two-link brachiating robot, IEEE Int. Conf. on Robotics&Automation, pp.787-792, (1998)

[18]Y. Hasegawa and T. Fukuda, Learning Method for Hierarchical Behavior Controller, IEEE Int. Conf. on Robotics&Automation, pp.2799-2804, (1999)

[19]Y. Hasegawa, Y. Ito and T. Fukuda, Behavior Coordination and its Modification on Brachiation-type Mobile Robot, IEEE Int. Conf. on Robotics&Automation, pp3984-3989, (2000)

[20]Y.Hasegawa, J. Matsuno and T.Fukuda, Regrasping behavior generation for Rectangular Solid Object, IEEE Int. Conf. on Robotics&Automation, pp3567-3572, (2000)